HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, Colorado 80527-2400

PATENT APPLICATION

ATTORNEY DOCKET NO. 10018215-1

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Michael S. Allison, et al.

Confirmation No.: 9960

Application No.: 09/917,377

Examiner: Khanh B. Pham

Filing Date: July 28, 2001

Group Art Unit: 2177

Title: METHOD FOR EXTRACTING, FILTERING, AND SEPARATING EVENTS FROM SYSTEM FIRMWARE AND SOFTWARE

Mail Stop Appeal Brief -- Patent
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL LETTER FOR RESPONSE/AMENDMENT

Sir:

Transmitted herewith is/are the following in the above-identified application:

( ) Response/Amendment
( ) New fee as calculated below
( ) No additional fee
(X) Other: Amended Appeal Brief & Return Post Card
( ) Petition to extend time to respond
( ) Supplemental Declaration

CLAIMS AS AMENDED BY OTHER THAN A SMALL ENTITY						
(1) FOR	(2) CLAIMS REMAINING AFTER AMENDMENT	(3) NUMBER EXTRA	(4) HIGHEST NUMBER PREVIOUSLY PAID FOR	(5) PRESENT EXTRA	(6) RATE	(7) ADDITIONAL FEES
TOTAL CLAIMS		MINUS		= 0	X $50	$ 0
INDEP. CLAIMS		MINUS		= 0	X $200	$ 0
[ ] FIRST PRESENTATION OF A MULTIPLE DEPENDENT CLAIM					+ $360	$ 0
EXTENSION FEE	1ST MONTH $120.00 [ ]	2ND MONTH $450.00 [ ]	3RD MONTH $1020.00 [ ]	4TH MONTH $1590.00 [ ]		$ 0
					OTHER FEES	$
					TOTAL ADDITIONAL FEE FOR THIS AMENDMENT	$ 0

Charge $ 0 to Deposit Account 08-2025. At any time during the pendency of this application, please charge any fees required or credit any overpayment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

"Express Mail" label no. EV413241332US

Date of Deposit Mar. 23, 2005

I hereby certify that this is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to: Commissioner for Patents, Alexandria, VA 22313-1450.

By Melissa R Smith

Typed Name: **Melissa Smith**

Respectfully submitted,

Michael S. Allison, et al.

By Curtis A. Vock

Curtis A. Vock

Attorney/Agent for Applicant(s)
Reg. No. **38,356**

Date: **Mar. 23, 2005**

Telephone No.: **(720) 931-3011**

Rev 12/04 (TnAmdExp)

- Attach as First Page to Transmitted Papers -

03-24-05

AF/2177

PATENT
Attorney Docket No.: 10018215-1
Express Mail Label No.: EV413241332US

Applicant(s):	Michael S Allison et al.	Confirmation No.	9960
Serial No.	09/917,377	Group Art Unit:	2177
Filed:	July 28, 2001	Examiner:	Homere, Jean Raymond
For:	Method for Extracting, Filtering, and Separating Events from System Firmware and Software	Attorney Docket No.	10018215-1

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

AMENDED APPEAL BRIEF

Dear Sir:

In accord with 37 CFR §1.192, and fully responsive to the Office Actions of April 23, 2004 and March 8, 2005 (Notice of Non-Compliance), Appellants hereby file their amended appeal brief in support of their Appeal in the above-identified matter. A notice of appeal, with appropriate fee of $330 as required by 37 CFR §1.17b, was filed on 28 June 2004. Three copies of this brief are enclosed, as required under 37 CFR §1.192(c). This amended appeal brief is filed within one month of the mailing of the Notice of Non-Compliance and specifically addresses the Examiner's objection pursuant to 37 C.F.R. §1.192(c)(5).

(1) **Real party in interest.**

The real party in interest for this appeal is Hewlett-Packard Development Company, L.P. (HPDC), a limited partnership established under the laws of the State of Texas and having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, L.L.C. Evidence of this assignment, which was recorded on September 30, 2003, may be found at reel/frame 014061/0492.

(2) **Related appeals and interferences.**

No other appeals or interferences are currently known to Appellant that will directly affect, be directly affected by, or have a bearing on the decision to be rendered by the Board of Patent Appeals and Interferences in the present appeal.

(3) **Status of claims.**

Claims 1-20 are pending in this application. Applicants appeal all claims 1-20. Claims 1-2, 4-20 now stand rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,467,052 ("Kaler"). Claim 3 stands rejected as being unpatentable over Kaler in view of Applicant's admitted prior art ("AAPA") in the background of the application, under to 35 U.S.C. §103.

(4) **Status of amendments.**

This application was filed on July 28, 2001. A first office action was mailed on December 11, 2003, to which a response (without amendment to the claims) was filed and entered February 23, 2004. On April 23, 2004, a final office action was mailed, prompting this appeal. The claims currently pending are original claims.

(5) **Summary of the invention.**

The inventions of claims 1-17 relate to a method for processing events from electronic architecture, the architecture of the type having one or more entities generating the events. This electronic architecture is for example

illustrated in FIG. 2 as item 104; see also paragraph 18, lines 1-7. The "events" are for example chassis logs, such as discussed in connection with paragraph 22, lines 105; see also FIG. 3A, 3B (and in particular steps 212, 218, 234). The "entities" are for example one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers, such as described in paragraph 2, lines 1-3, of the specification. The method steps of these claims 1-17 are thus illustrated in exemplary fashion in FIG. 3A, 3B, 4A, 4B and discussed throughout paragraphs 19, 21-26.

The inventions of claims 18-20 relate to a system that processes events from electronic architecture. Illustratively, this architecture is shown as item 104, FIG. 2. "Events" are for example chassis logs. See paragraph 22, lines 105; see also FIG. 3A, 3B, and in particular steps 212, 218, 234. "Entities" are for example one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers. See paragraph 2, lines 1-3. The system of claims 18-20 includes an extraction tool that is for example illustrated in FIG. 2 as GETCC 102 and analyzers 120. The interface between the extraction tool and architecture 104 (or log file 106) is for example illustrated in FIG. 2, and in Table 1 on pages 4 and 5 of the specification. See also method steps of GETCC and analyzers within FIG. 3A, 3B, 4A, 4B.

(6) Issues.

A. Whether claims 1-2, 4-20 are anticipated by U.S. Patent No. 6,467,052 ("Kaler") in accordance 35 U.S.C. §102(e).

B. Whether claim 3 is patentable over Kaler in view of AAPA in accordance with 35 U.S.C. §103.

(7) Grouping of claims.

Group I consists of claims 1-2, 4-20. The claims of Group I do not stand or fall together. Group II consists of claim 3.

(8) Arguments.

ARGUMENT A

Kaler does not teach or suggest each and every claim limitation within claims 1-2, 4-20 as required by 35 U.S.C. §102(e).

To anticipate a claim, Kaler must teach every element of the claim and "the identical invention must be shown in as complete detail as contained in the ... claim." *MPEP 2131* citing *Verdegaal Bros. V. Union Oil Co. of California,* 814 F.2d 628, 2 USPQ2d 1051 (Fed. Cir. 1987) and *Richardson v. Suzuki Motor Co.,* 868 F.2d 1226, 9 USPQ2d 1913 (Fed. Cir. 1989). Kaler does not teach every element of claims 1, 2 and 4-20.

Claim 1 recites a method for processing events from electronic architecture, the architecture of the type having one or more entities generating the events, and requires the following steps:

(A) extracting the events from the architecture;

(B) separating the events according to the entities; and

(C) transforming the events to one or more text strings.

Kaler does not teach or disclose these steps. The Examiner cites Kaler, col. 3, line 66 – col. 4, line 12 with the proposition that Kaler discloses step (A); however, a close review of this section reveals that Kaler requires an in-process event creator (IEC) that *creates* an event desired by a developer. See also Kaler col. 4, lines 17-24. This is not the same as "extracting" an event previously generated by electronic architecture, as in claim 1 (and as clearly supported in Applicants' specification by, for example, the getcc processing section 102, FIG. 2).

Once Kaler generates the events, filter reduction techniques are employed to extract only information of interest from the events, to reduce the performance impact of monitoring. See Kaler, col. 4, lines 51-56. By way of example, col. 16, line 1-col. 17, line 25 of Kaler discloses how events may be predefined or custom-defined to assist in analysis (e.g., to assist in analyzing relationships between events). In particular, Kaler teaches extracting events from the software architecture to separate events according to fields, types and categories. Kaler, col. 16, line1 - col. 17, line 25.

This is not, however, the same as "extracting" events, as in step A of claim 1; nor is it the same as separating events according to entities (step B). Applicants' events are internally generated by entities of electronic architecture. In paragraph

[0008] of Applicants' specification, for example, such extraction is described in the context of an extraction tool that "connects with electronic architecture to *extract* and analyze *internally generated events*" (emphasis added). There are no "event creators" as in Kaler. The events of claim 1 are preexisting (internally generated by electronic architecture, for example) and therefore have to be extracted (step A).

Consider, for example, claim 3, which depends from claim 1 and which specifically claims that the extraction of events (step A) means the extraction of "chassis logs." Chassis logs are events internally generated by connected electronic architecture and contain encoded data of electronic architecture information. See Applicants' specification, paragraph [0003]. As such, chassis logs are for example produced by intelligent hardware modules, I/O drivers, processors (local and additional), the operating system, and system firmware; they contain information of events pertaining to operation and status of electronic architecture.

Kaler does not, at all, disclose utilization or processing of chassis logs, as in claim 3. Rather, Kaler teaches event creation and then monitoring "...the executing process for particular situations that occur which the developer wants to be monitored," for creating "an "event" that can be captured and later analyzed." Kaler, col. 4, lines 5-8. Therefore, Applicants' claim 1 (and similarly claim 3) patentably differ from Kaler in that, for example, there are no event creators; events are instead extracted from entities of electronic architecture.

Claim 1 (step B) also requires separating the events according to the entities of the electronic architecture. Kaler does not teach or suggest the separation of events according to entities of electronic architecture; rather, Kaler teaches a Local Event Concentrator (LEC) that collects and stores events of the executing process for particular situations. This LEC sends events to a developer who specifies "by means of a 'filter' what to look for in the system under examination." See Kaler, col. 4, lines 13-14. Kaler's system thus filters events according to event type, but does not separate events according to entity, as in claim 1.

For at least the reasons discussed above, Applicants contend that Kaler does not anticipate claim 1; therefore, claim 1 is patentable over Kaler.

Claims 2 and 4-17 depend from claim 1 and benefit from like arguments. In addition, however, these claims provide other features that patentably distinguish over Kaler, including, for example, the following steps:

- the step of filtering the events (claim 2). In the context of claim 1, Kaler does not filter events. Again, Kaler's IEC generates the events and does not then require filtering.
- the step of coupling a getcc extraction tool to the architecture (claim 4). Kaler is silent as to such an extraction tool. In fact, Kaler *teaches away* from extraction as in step A of claim 1 by generating (not extracting!!) the events with event creators. See Kaler col. 4, lines 17-24. Respectfully, we do not believe that the Examiner's statement of "telnet" in the context of Kaler's figure 1 is accurate. In particular, irrespective of modem 54 shown in figure 1 of Kaler, Kaler does not disclose, teach or suggest extracting events, as argued above in connection with claim 1.
- the step of coupling comprises utilizing telnet (claim 5). Again, we contend that Kaler does not, in the context of claims 1 and 4, teach utilizing telnet.
- the step of extracting events from the server (claim 6). Kaler does not teach extraction of events; it instead discloses generating events. Kaler therefore also does not disclose extracting events from a server.
- the step of converting a binary representation of the events to the text strings (claim 7). Kaler does not even use the word "binary" within its text. Applicants' claim 7 thus further distinguishes over Kaler, for example, in that events internally generated may be extracted in binary form. Kaler instead creates events; Kaler's generated events appear entirely non-binary. See, e.g., Kaler Table 3, col. 17. Respectfully, the Examiner's reference to Kaler, col. 19, lines 13-22, also does not disclose binary conversion.
- analyzing the text strings and producing a human interpretable statement summarizing at least one of the events associated with the

text strings (claim 8). Interpretive event summaries are not disclosed by Kaler, which instead discloses creation of custom fields. See Kaler, col. 15 and 16. Respectfully, the Examiner's reference to Kaler, col. 19, lines 50-56, does not teach producing human interpretable statements. This section in fact teaches away from claim 8 since, for example, Kaler discloses that users do not need to understand the very events generated by Kaler.

- wherein the entities comprises one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers (claim 9). In the context of claim 1, Kaler does not teach such entities and the extraction of events from such entities.
- controlling one or more steps of extracting, separating and transforming via one or more command line options (claim 10). Kaler is silent as to the use of command line options. The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we disagree. Claim 10 patentably distinguishes over Kaler since, among other reasons, Kaler fails to teach command line options as in claim 10.
- controlling one or more steps of extracting, separating and transforming according to one or more configuration files (claim 11). The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we again disagree. Claim 11 patentably distinguishes over Kaler since, among other reasons, Kaler fails to teach configuration files as in claim 11 (the word "configuration" is not even mentioned in Kaler).
- inputting the command line options via a graphical user interface (claim 12). The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we again disagree. Claim 12 patentably distinguishes over Kaler since, among other reasons, Kaler fails to

teach command line options and graphical user interface as in claim 12.

- updating the command line options automatically from the architecture (claim 13). The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we again disagree. Claim 13 patentably distinguishes over Kaler since, among other reasons, Kaler fails to teach command line options and graphical user interface as in claim 12.
- specifying one or more cells of the architecture, and extracting the events only from the one or more cells (claim 14). The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we again disagree. Claim 14 patentably distinguishes over Kaler since, among other reasons, Kaler fails to teach extracting events from cells of electronic architecture, as in claim 14.
- specifying one or more processors of the architecture, and extracting the events only from the one or more processors (claim 15). The Examiner makes an extremely broad and generalized assertion that claims 10-20 "have been discussed in the rejection of claims 1-2, 4-9" – but we again disagree. Claim 15 patentably distinguishes over Kaler since, among other reasons, Kaler fails to teach extracting events only from processors as in claim 15.
- saving a log file representative of the events (claim 16). In the context of claim 1, from which claim 16 depends, Kaler does not teach saving a log file representative of the events.
- transmitting the text strings to one or more analyzers associated with one or more entities and analyzing the text strings at the one or more analyzers (claim 17). In the context of claim 1, from which claim 17 depends, Kaler does not teach saving a log file representative of the events.

Claim 18 is a system for processing events from electronic architecture of the type having one or more entities generating the events. Claim 18 requires the following elements:

(A) an extraction tool for extracting the events from the architecture, separating the events according to the entities, and transforming the events to one or more text strings; and

(B) an interface for coupling the extraction tool to one or more of the architecture and a log file storing the events from the architecture.

The Examiner broadly rejects, without specificity, claims 18-20 in paragraph 4 of the final office action without additional argument and based on previous rejections of claims 1, 2, 4-9. Respectfully we disagree since, among other reasons, Kaler does not teach the elements of independent claim 18. We have already shown and argued in connection with claim 4 that Kaler does not disclose an extraction tool, though this element is required in claim 18. Therefore, among other reasons, Kaler cannot anticipate claim 18. Once again, Kaler discloses generating events, not "extracting" events as required by claim 18. Reconsideration of claim 18 is requested.

Claims 19-20 depend from claim 18 and benefit from like arguments. These claims also have other features that patentably distinguish over Kaler. For example, claim 19 discloses "chassis logs," argued above, and below, in connection with claim 3. Kaler is silent to chassis logs and cannot, therefore, anticipate claim 19. The word "chassis" is not even mentioned within Kaler. In claim 20, Kaler again does not disclose processing events into human interpretive statements, as previously argued in connection with claim 8.

ARGUMENT B

Kaler and AAPA do not teach or suggest each and every claim limitation within claim 3 as required by 35 U.S.C. §103.

Applicants contend that the cited art does not render Group II claim 3 *prima facie* obvious. The following is a quotation of from the MPEP setting forth the three basic criteria that must be met to establish a *prima facie* case of obviousness:

> To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references

themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the references or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. MPEP, §2142, citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Claim 3 describes "extracting chassis logs, wherein the step of separating the events comprises separating the chassis logs, and wherein the step of transforming events comprises transforming the chassis logs to text strings." Claim 3 depends from claim 1 and benefits from previous arguments made in connection with claim 1. Applicants have shown that Kaler does not teach each of the limitations of claim 1. We have also argued that Kaler is silent to the processing or utilization of chassis logs, as in claim 3. Therefore, Kaler alone cannot teach or suggest the limitations of claim 3, as required under 35 U.S.C. §103.

The AAPA – disclosed in the background of the present specification - also does not teach or suggest every element of Claim 3. The AAPA merely discloses that chassis logs are (a) generated by entities of a system during operation of the system and (b) not easily assessed, requiring an engineer intimate with the system and chassis logs to decode each chassis log to identify problems in the electronic architecture. Accordingly, Kaler and/or AAPA do not teach or suggest every element of Applicants' claim 3, as required under 35 U.S.C. §103. There is also no motivation to combine Kaler and the AAPA since, for example, Kaler teaches generation of events and not (a) the extraction of events (claim 1) nor (b) the utilization of chassis logs from electronic architecture (claim 3). Since neither Kaler nor the AAPA disclose extracting events – e.g., chassis logs – from electronic architecture, as required in claim 3, claim 3 too patentably distinguishes over Kaler and the AAPA.

(9) Appendix.

Appellants enclose a copy of the claims involved in this appeal as an appendix hereto.

(10) Conclusions.

Applicants respectfully submit that the claims Groups I-II patentably distinguish over the art of record. Other than the costs for the notice of appeal and

appeal brief, no further fees are deemed due in connection with this matter. However, the Commissioner is hereby authorized to charge any fees which may be due in this matter from Deposit Account Number 08-2025.

Respectfully submitted,

Curtis A. Vock

Curtis A. Vock, Reg. No. 38,356
LATHROP & GAGE LC
4845 Pearl East Circle, Suite 300
Boulder, Colorado 80301
(720) 931-3011 Telephone
(720) 931-3001 Facsimile

APPENDIX TO APPEAL BRIEF

1. (Original) A method for processing events from electronic architecture, the architecture of the type having one or more entities generating the events, comprising the steps of:

extracting the events from the architecture;

separating the events according to the entities; and

transforming the events to one or more text strings.

2. (Original) A method of claim 1, further comprising the step of filtering the events.

3. (Original) A method of claim 1, wherein the step of extracting the events comprises extracting chassis logs, wherein the step of separating the events comprises separating the chassis logs, and wherein the step of transforming events comprises transforming the chassis logs to text strings.

4. (Original) A method of claim 1, further comprising the step of coupling a getcc extraction tool to the architecture.

5. (Original) A method of claim 4, wherein the step of coupling comprises utilizing telnet.

6. (Original) A method of claim 1, the architecture being a server, and wherein the step of extracting events from the architecture comprises extracting events from the server.

7. (Original) A method of claim 1, wherein the step of transforming comprises converting a binary representation of the events to the text strings.

8. (Original) A method of claim 1, further comprising the step of analyzing the text strings and producing a human interpretable statement summarizing at least one of the events associated with the text strings.

9. (Original) A method of claim 1, wherein the entities comprises one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers.

10. (Original) A method of claim 1, further comprising the step of controlling one or more steps of extracting, separating and transforming via one or more command line options.

11. (Original) A method of claim 10, further comprising controlling one or more steps of extracting, separating and transforming according to one or more configuration files.

12. (Original) A method of claim 10, wherein the step of controlling comprises inputting the command line options via a graphical user interface.

13. (Original) A method of claim 10, wherein the step of controlling comprises updating the command line options automatically from the architecture.

14. (Original) A method of claim 1, further comprising specifying one or more cells of the architecture, and extracting the events only from the one or more cells.

15. (Original) A method of claim 1, further comprising specifying one or more processors of the architecture, and extracting the events only from the one or more processors.

16. (Original) A method of claim 1, further comprising the step of saving a log file representative of the events.

17. (Original) A method of claim 1, further comprising the steps of transmitting the text strings to one or more analyzers associated with one or more entities and analyzing the text strings at the one or more analyzers.

18. (Original) A system for processing events from electronic architecture, the architecture of the type having one or more entities generating the events, comprising:

an extraction tool for extracting the events from the architecture, separating the events according to the entities, and transforming the events to one or more text strings; and

an interface for coupling the extraction tool to one or more of the architecture and a log file storing the events from the architecture.

19. (Original) A system of claim 18, wherein the entities comprise one or more of firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers, and wherein the events comprise chassis logs from one or more of the firmware, software, processors, architecture monitors, power monitors, cabinet monitors, and I/O drivers.

20. (Original) A system of claim 18, further comprising one or more analyzers coupled to the extraction tool, the analyzers processing the text strings into one or more human interpretable statements summarizing at least one of the events associated with the text strings.